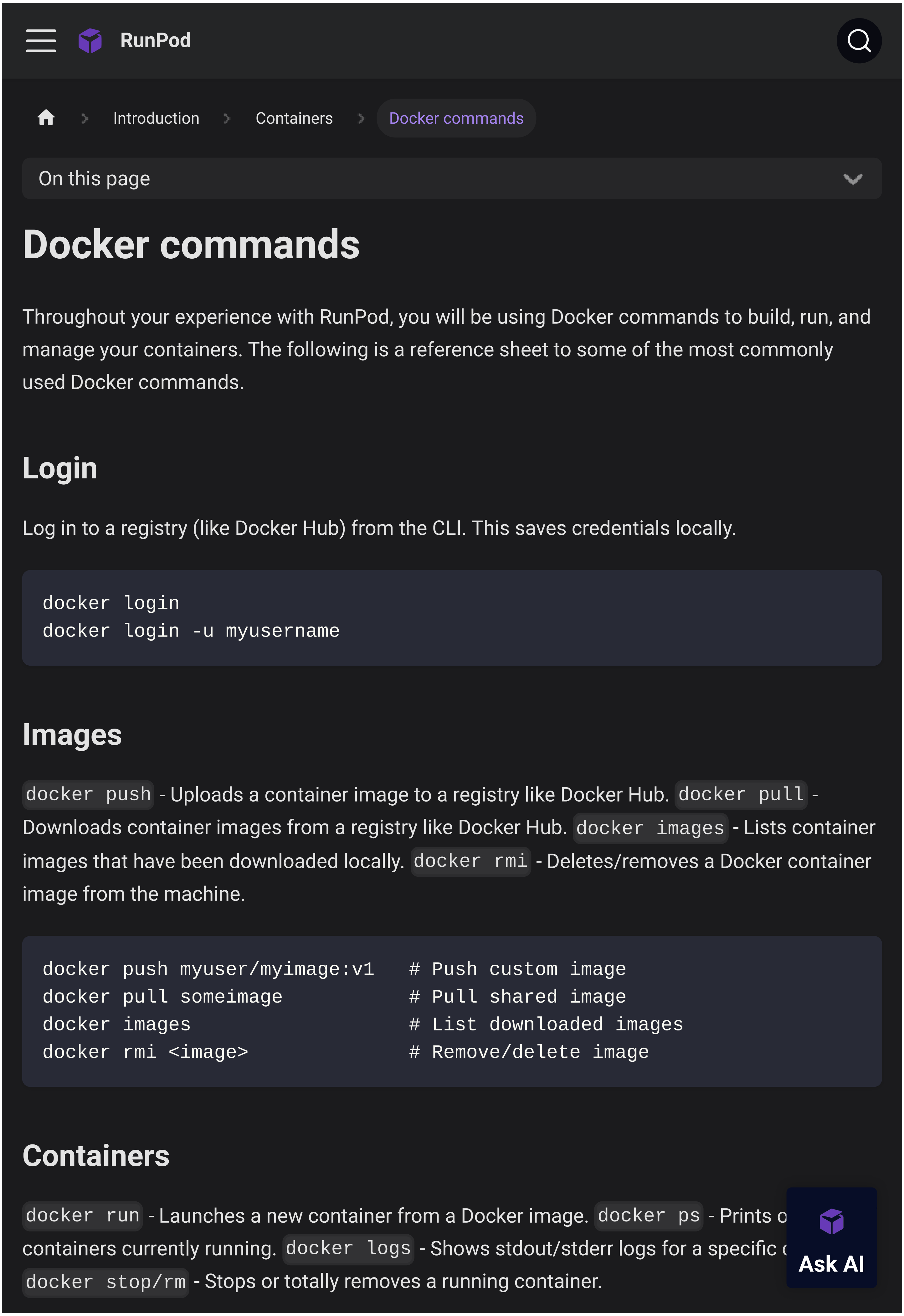

Docker commands

Throughout your experience with RunPod, you will be using Docker commands to build, run, and manage your containers. The following is a reference sheet to some of the most commonly used Docker commands.

Login

Log in to a registry (like Docker Hub) from the CLI. This saves credentials locally.

```
docker login
docker login -u myusername
```

Images

`docker push` - Uploads a container image to a registry like Docker Hub. `docker pull` - Downloads container images from a registry like Docker Hub. `docker images` - Lists container images that have been downloaded locally. `docker rmi` - Deletes/removes a Docker container image from the machine.

```
docker push myuser/myimage:v1   # Push custom image
docker pull someimage           # Pull shared image
docker images                   # List downloaded images
docker rmi <image>              # Remove/delete image
```

Containers

`docker run` - Launches a new container from a Docker image. `docker ps` - Prints o containers currently running. `docker logs` - Shows stdout/stderr logs for a specific c `docker stop/rm` - Stops or totally removes a running container.

```
docker run        # Start new container from image
docker ps         # List running containers
docker logs       # Print logs from container
docker stop       # Stop running container
docker rm         # Remove/delete container
```

Dockerfile

`docker build` - Builds a Docker image by reading build instructions from a Dockerfile.

```
docker build                           # Build image from Dockerfile
docker build --platform=linux/amd64   # Build for specific architecture
```

> ⓘ **NOTE**
>
> For the purposes of using Docker with RunPod, you should ensure your build command uses the `--platform=linux/amd64` flag to build for the correct architecture.

Volumes

`docker volume create` - Creates a persisted and managed volume that can outlive containers. `docker run -v` - Mounts a volume into a specific container to allow persisting data past container lifecycle.

```
docker volume create        # Create volume
docker run -v <vol>:/data   # Mount volume into container
```

Network

`docker network create` - Creates a custom virtual network for containers to communicate over. `docker run --network=<name>` - Connects a running container to a Docker user-defined network.

```
docker network create           # Create user-defined network
docker run --network=<name>     # Connect container
```

Ask AI

Execute

`docker exec` - Execute a command in an already running container. Useful for debugging/inspecting containers:

```
docker exec
docker exec mycontainer ls -l /etc      # List files in container
```

Edit this page

Previous
« Persist data outside of containers

Next
Run your first Fast Stable Diffusion with Jupyter Notebook »

Docs

Overview

Tutorials

AI APIs

Community

Discord

Contact us

More

Blog

GitHub

Copyright © 2024 RunPod

Ask AI